Bolt Beranek and Newman Inc.

(11)

Report No. 5799

Research in Continuous Speech Recognition

Final Report

1 October 1981-15 November 1984

December 1984

Prepared for:
Advanced Research Projects Agency
and
Office of Naval Research

Report No. 5799

RESEARCH IN CONTINUOUS SPEECH RECOGNITION

Final Report
1 October 1981 to 15 November 1984

December 1984

Principal Investigator:
John Makhoul
(617) 497-3332

Prepared for:

Defense Advanced Research Projects Agency
and
Office of Naval Research

ARPA Order Nos. 4311/4707

Contract No. N00014-81-C-0738

Effective Date of Contract:
1 October 1981

Contract Expiration Date:
15 November 1983

This research was supported by the Advanced Research Projects Agency of the Department of the Defense and was monitored by ONR under Contract No. N00014-81-C-0738. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: BBN Report #5799	2. GOVT ACCESSION NO.: AD-A150 424
	3. RECIPIENT'S CATALOG NUMBER:
4. TITLE (and Subtitle): RESEARCH IN CONTINUOUS SPEECH RECOGNITION	5. TYPE OF REPORT & PERIOD COVERED: Final Report 1 Oct 81 - 15 Nov 84
	6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): Richard Schwartz, Yen-Lu Chow, Owen Kimball, Michael Krasner, John Makhoul, Salim Roucos	8. CONTRACT OR GRANT NUMBER(s): N00014-81-C-0738
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Bolt Beranek and Newman Inc. 10 Moulton Street Cambridge, MA 02238	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:
11. CONTROLLING OFFICE NAME AND ADDRESS: Office of Naval Research Department of the Navy Arlington, VA 22217	12. REPORT DATE: December 1984
	13. NUMBER OF PAGES: 27
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office):	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report)

Distribution of the document is unlimited. It may be released to the Clearinghouse, Dept. of Commerce, for sale to the general public.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Speech Recognition, Phonetic Recognition, Hidden Markov Models, Phonetic Context, Acoustic-Phonetic Features, Context-Dependent Phonetic Models.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This final report describes the results of a three-year project in an ongoing basic research effort to develop techniques for the automatic recognition of speech sounds (phonemes) in unrestricted continuous speech. A basic working phonetic recognition system has been designed, implemented on a VAX 11/780, and tested. At over 75% phonetic accuracy, the results have already exceeded previously-published phonetic recognition results on continuous speech for a single speaker. Our future goal will be to increase the recognition accuracy to over 80% for many speakers.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

	Page
1. EXECUTIVE SUMMARY	1
2. INTRODUCTION	3
2.1 Phonetic Recognition	3
2.2 Approach	4
2.3 Review of Problem	4
2.3.1 Phonetic Context	4
2.3.2 Spectral vs Acoustic-Phonetic Features	5
2.4 Outline	6
3. PHONETIC HIDDEN MARKOV MODEL	7
4. MODELING PHONEMES IN CONTEXT	9
4.1 Triphone Context Model	9
4.2 Combining Models	9
4.3 Interpolated Models	10
5. TRAINING AND RECOGNITION SYSTEMS	12
5.1 Analysis	12
5.2 Variable-Frame-Rate	12
5.3 Training	12
5.4 Recognition	13
5.5 Search Strategies	14
5.5.1 Best-First Search	15
5.5.2 Time-Synchronous Beam-Search	16
6. EXPERIMENTS WITH E-SET	18

7. EXPERIMENTS WITH CONTINUOUS SPEECH 21

8. ACOUSTIC-PHONETIC FEATURES 23

9. CONCLUSION 25

LIST OF FIGURES

FIG. 1. Hidden Markov Model of a Phoneme. 7
FIG. 2. E-set recognition performance 19

LIST OF TABLES

TABLE 1. Weight for Left Context Model 20
TABLE 2. Phonetic recognition accuracy for continuous speech. (PH = phoneme model; L = left-context model; R = right-context model;) 22

1. EXECUTIVE SUMMARY

This final report describes the results of a three-year project in an ongoing basic research effort to develop techniques for the automatic recognition of speech sounds (phonemes) in unrestricted continuous speech. A basic working phonetic recognition system has been designed, implemented on a VAX 11/780, and tested. At over 75% phonetic accuracy, the results have already exceeded previously-published phonetic recognition results on continuous speech for a single speaker. Our future goal will be to increase the recognition accuracy to over 80% for many speakers.

Our first major milestone was reached in August 1983 when we completed our initial phonetic recognition system. The system was based on a hidden Markov model (HMM) of speech spectral parameter movements in each phoneme. The HMM is a flexible mathematical tool that is especially suited to modeling variabilities in time and space (in this case, frequency spectrum). The major innovation in our work was that the basic HMM algorithm was capable of including the effects of left and right contexts in modelling each phoneme. We have demonstrated that our context-dependent models result in significantly higher recognition accuracy than context-independent models. Furthermore, we have developed methods that determine the extent to which context should be included, based on the amount of speech data available to train the system. Another important aspect of the work is that the training procedures we have used are largely automatic and require little human interaction.

In June 1984 we completed another major milestone in designing and implementing a capability to incorporate acoustic-phonetic knowledge (in the form of acoustic-phonetic features) into a probabilistic hidden Markov formalism. In our system, the categorical decisions usually associated with heuristic acoustic-phonetic algorithms are replaced by automatic training techniques and global search strategies. The acoustic-phonetic features are expected to improve the system's ability to make fine phonetic distinctions. To test this new system capability, we added features that help in discriminating among the unvoiced plosives [p, t, k]. (The features measured the frequency and energy level of the plosive burst.) The recognition rate for the unvoiced plosives in continuous speech increased from 61% when using the spectral HMMs to 85% after the burst features were included. The recognition accuracy of the other phonemes was not affected in the process. We expect the overall performance to improve further by the incorporation of large numbers of additional acoustic-phonetic features.

Finally, a considerable effort has enabled us to reduce the computation, virtual address space, and file storage needed for the various programs and data structures by at least an order of magnitude each. This has made it possible to perform significantly more experiments than were previously possible. In particular, our new time-synchronous, pruned search for phoneme sequences has resulted in two orders of magnitude increase in speed over best-first search algorithms.

2. INTRODUCTION

2.1 Phonetic Recognition

Automatic speech recognition for constrained applications is now feasible. The constraints may be a limitation on the vocabulary size, the branching factor of the grammar, the number of speakers, or a requirement for isolated words. A major breakthrough in the ability to distinguish between similar words reliably is needed before unconstrained speech recognition can be achieved. Therefore the primary emphasis in our research in recent years has been in recognizing the phonemes in continuous speech without the aid of a constraining factor such as a phonetic dictionary. Only after adequate phonetic recognition performance has been achieved, would we attempt to build a large scale, unconstrained speech recognition system.

While there are as many measures of phonetic recognition accuracy as there are phonetic recognizers, it is clear that minimum accuracies of 80%–90% are necessary to support large scale speech recognition. Our experiments with phonetic vocoders showed that human listeners could understand the output speech only if at least 80% of the phonemes were correctly recognized [1]. Also, extrapolation from phonetically based speech recognition systems would predict that about 90% correct phonetic recognition may be necessary if vocabulary and grammatical constraints are weak [2, 3].

That speech recognition should be based on phonetic recognition may not be immediately clear. Indeed, for moderate vocabularies of a few hundred words, where training all the words is possible, it has been shown that recognition based on a word model achieves higher performance [4]. All commercially available speech recognizers also use word models. However, for very large vocabularies, it is not practical to ask each user of a speech recognition system to say each of the words in many different contexts. If the system were intended to be speaker–independent, the one–time cost of having many speakers say all the words may be acceptable, but there still remains the problem of predicting phonological variability both within the word and between words, which may be more easily done using a phoneme model for words. Also, the added variability due to multiple speakers might make the recognition problem too difficult. For these reasons, it is generally assumed that the many words would be modeled in terms of their phonetic spellings. A new speaker would first read enough speech for the system to adapt the models of the phonemes to that speaker. Then,

the system would use the speaker-dependent set of phonetic models with the phonetic dictionary to model the words.

2.2 Approach

Our basic approach to phonetic recognition can be characterized as combining multiple sources of knowledge to achieve the advantages of each knowledge source. We have applied this principle in three areas related to phonetic recognition: the modeling of phonetic context, the combined use of spectral features and acoustic-phonetic features, and the application of probabilistic phonotactic constraints.

2.3 Review of Problem

In this section we discuss the particular problems that we are attempting to solve.

2.3.1 Phonetic Context

If the basic acoustic model used for speech recognition represents the phoneme, the implicit assumption is made that the acoustic realization of the phoneme is independent of the phonetic context in which it appears. Of course, we know that phonemes are affected significantly - particularly near the transitions - by the neighboring phonemes. There have been several attempts to account for the effects of phonetic context. The HMM approach can account for this effect by modeling each part of the phoneme separately. Thus, the probability density function (pdf) that represents the beginning and end of the phoneme will have a wider variance, or (in the case of a discrete pdf) allow for more possible spectra than will the middle part of the phoneme, which is less affected by phonetic context. However, this widening of the pdfs at the phoneme transitions does not correctly reflect the conditional relation between the acoustics of a phoneme and its neighboring phonemes.

As a result, there have been many suggestions to model the acoustics of units larger than phonemes, such as diphones [5, 6], demisyllables [7], syllables [8, 9], etc. The longer acoustic units implicitly account for the coarticulatory effects of the phonemes within the unit on each other. For example, a model of the syllable unit implicitly accounts for the effect of consonants on the vowel in the same syllable. A diphone unit models the transition between the two phonemes. The coarticulatory

effects between acoustic units are assumed to be negligible. In our work, we use a model of the phoneme conditioned on the phonetic context, which explicitly takes into account the effects of context.

The major difficulty in modeling contextual effects is that the number of these longer units or contexts of interest is very large. For instance, there are about 2,500 diphones and about 10,000 syllables in English. Therefore, it is impossible to gather detailed statistics about the likely acoustic realizations of all of them from a reasonable sized data base. In a large database (say one half hour) of natural speech, more than half of the diphones, and most of the possible syllables will not occur even once. How, then, can we model those coarticulatory effects which we believe to be important for speech recognition? We describe our approach to this problem in Section 4.

2.3.2 Spectral vs Acoustic-Phonetic Features

Most template matching systems use a model of the short-term power spectrum as the basic representation of the speech. The motivation for using spectral features for speech recognition is simple. It is well known that intelligible speech can be recreated from a sequence of quantized power spectra. Therefore, the distinction between different words must be contained in the sequence of spectra. It is also quite easy to state the speech recognition problem as a straightforward decoding problem using well-known communication theory principles [3]. However, there is the widespread belief that other "speech motivated" features should be more useful. These features, which we call acoustic-phonetic (AP) features, measure specific aspects of the signal spectrum at particular locations within phonemes, and are designed for making specialized phonetic distinctions. The canonical example of such a feature in the field of vision is the distinction between the letters "O" and "Q". In speech, we know features that occupy a very small part of the time-frequency plane contain most of the information that is needed to make certain distinctions. For example, in an unvoiced plosive, the spectrum during the silence is of no value in distinguishing among the phonemes /p, t, k/. The spectrum during the aspiration is mostly a function of the following vowel, and depends also on whether the plosive is preceded by an /s/. However, the characteristics of the burst and the directions of the formant transitions are quite indicative of the plosive (although there is some effect due to phonetic context).

Despite the attraction of these AP features, they suffer from the fact that they

are an incomplete set. That is, they cannot be used to reconstruct the speech, and therefore, are not guaranteed to contain all the information necessary to understand the speech. In this paper, we will present arguments why AP features can still be of some beneficial use, and suggest an accommodation of these two approaches that takes advantage of the merits of each.

2.4 Outline

Section 3 defines and describes the hidden Markov model for a phoneme that we use for phonetic recognition. Section 4 discusses the problem of modeling phonetic context and proposes a solution that overcomes the training issues typically associated with solutions for this problem. The training and recognition procedures used in all our experiments are described in Section 5. Sections 6 and 7 present the results of phonetic recognition experiments for the E-set problem and for unconstrained continuous speech. Section 8 contains a discussion of the combined use of spectral features and acoustic-phonetic features within the same hidden Markov formalism. Finally, Section 9 contains some conclusions that we feel can be drawn from this research, and our plan for the immediate future.

3. PHONETIC HIDDEN MARKOV MODEL

We have chosen to use a hidden Markov Model (HMM) to represent the time variations of a phoneme. We give here a brief description of the use of HMM's. For a more complete mathematical review of HMM's, see [10]. Figure 1 illustrates the HMM that we use. The most natural way to explain a HMM is to think of it as a model for synthesizing speech [11].

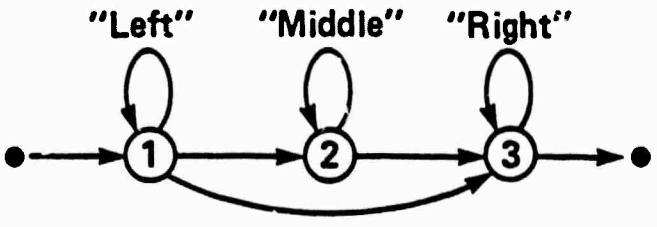

FIG. 1. Hidden Markov Model of a Phoneme.

In this model there are 5 states represented by circles. As with a Markov chain, there is associated with each combination of two states a transition probability (a_{ij}) which is the probability of going to state j given that the process is in state i. The arrows between states indicate the transitions that we allow (their probabilities are nonzero). Unlike a Markov chain, in which each state has associated with it a single output, each pdf-state of a HMM has an output probability density ($\underline{b}_j$). The output density gives the probability of each possible output symbol or vector given that the process is in that state. The three large open circles are pdf-states of the model. The small filled circles are called the initial and terminal states, and do not produce any output.

When using a HMM for synthesis, we first choose at random, according to some distribution, the state in which we will start. In our speech model, we must start with the initial state. Then, given the state i, we pick the next state j according to the

transition probabilities (a_{ij}), for all j. Once we have chosen a next state, we choose at random, among the possible outputs for this new state, according to its output probability density ($\underline{b}_i$). And so on. In this manner, the HMM will create a sequence of output symbols.

Once we have specified the form of a HMM, there are several interesting problems that we can solve related to this model. If we have a sequence of output vectors (observations) that we are told came from this HMM, we can estimate the parameters of the HMM (the a_{ij}'s and the $\underline{b}_i$'s) using an iterative procedure called the Baum-Welch or forward-backward algorithm [12]. Given a HMM with estimates of parameters and a sequence of observations, there are several computations one can perform. Using the Viterbi algorithm [13], one can efficiently identify and compute the probability of the most likely sequence of states to have produced the observed sequence. This (dynamic programming) algorithm requires computation that is linear in the number of nonzero transitions, and linear in the length of the observation sequence. A similar algorithm can be used to compute the probability that the entire HMM (all paths through the HMM) would have produced the observed sequence. The only difference between the two algorithms, is that when two paths come together, their probabilities are added, instead of keeping only the maximum probability. Using the probability of the output sequence given each HMM, one can also determine which of several HMM's is most likely to have produced an observed sequence. Finally, in the problem we wish to solve, we assume that a given observed sequence was produced by some sequence of HMM's (for example with each phoneme represented by one HMM), and we wish to determine the most likely sequence of those HMM's to have produced the observed output.

This last problem requires a search algorithm, since the number of possible HMM sequences grows exponentially with the length of the sequence. We compare the benefits of two possible suboptimal search algorithms in a later section.

4. MODELING PHONEMES IN CONTEXT

The different acoustic-phonetic units that have been proposed to account for phonetic context are, in fact, just trying to model the coarticulatory effects of adjacent phonemes on each other. There is not necessarily any significant importance in the units themselves. Therefore we have chosen to return to a model of the acoustics of each phoneme, but to take into account explicitly the phonetic context in which it appears.

4.1 Triphone Context Model

As an approximation to modeling the phoneme in all possible phonetic contexts, we have decided to take into account the immediately preceding and following phonemes. We call this a triphone context model, although it is really only a model for the middle phoneme conditioned on the two adjacent phonemes. It is expected that this model should account for almost all acoustic effects that are due to phonetic context. However, as discussed in the introduction, using a more complex unit results in a severe training problem, since there are many such context-dependent units, and therefore no longer enough tokens of each to develop a robust acoustic model. However, since some sequences occur much more often than others, there *are* enough samples of the more commonly occurring triphone contexts. For those triphone contexts that have not occurred a sufficient number of times, we could use the model for the phoneme that depends on the phoneme to the left ("left context model") combined with the model that depends on the phoneme to the right ("right context model"). For those left or right contexts that have not occurred, we can use the model for the phoneme that is independent of context ("phoneme model"). Thus a simple algorithm could choose the context model that is best for modeling the phoneme in triphone context depending on the number of training samples of each such model.

4.2 Combining Models

Rather than choosing among several models, it can be more effective to combine the models. For example, the first part of a phoneme is highly dependent on the phoneme to the left. Thus it is advantageous to take into account the left context model with as few as one or two tokens of that context in the training set. In contrast, the middle and last parts of the phoneme are less affected by the left context and, therefore, the model for these parts of the phoneme should not consider

the left context model unless there are a very large number of tokens of that context.

In addition to the phoneme model, the left and right context models, and the triphone model, we can also consider models that are dependent on a class of phonemes on one side or the other. For example, the effects of /p/ to the left of /a/ could be approximated by a combination of the effects due to preceding labial consonants and the effects due to preceding unvoiced plosives. These different models are combined with weights that depend on the type of model, the location within the middle phoneme, the number of tokens of the model observed in the training, and the types or classes of phonemes involved.

4.3 Interpolated Models

The solution of combining several different context-dependent models with weights that vary according to several factors leaves us with the problem of determining the weights to use for each combination of factors. We consider here three possible solutions: manually derived weights based on intuition, automatically derived weights based on a modeling of the training data, and manually or automatically derived weights based on a series of recognition experiments.

While it sounds like a difficult problem to generate a matrix of weights for all possible combinations of relevant factors, it was found to be quite straightforward (requiring about 1 hour), to arrive at a reasonably consistent set of weights that exhibited the desired behaviors.

The automatic solution we considered was a "Deleted Estimation" technique, using the forward-backward algorithm. This procedure has been used to smooth the transition probabilities of the stochastic grammar in the IBM speech recognition system [3]. The procedure entails dividing the training data into two groups. The first set is used to estimate the pdf's using the forward-backward algorithm (using an initial set of weights for combining models). Then, the second set of data is processed by the forward-backward algorithm, but in this case, the pdf's are kept fixed and the weights are optimized. This procedure may be repeated until the weights converge. If training data is limited, a jackknifing procedure can be used. The forward-backward algorithm will determine the weights that maximize the predicted probability of the second training set, given the pdf estimates derived from the first. In this way, the weights are derived to maximize the robustness of the combined models for the

modeling problem. Thus it is hoped that the weights will reflect the importance of the different pdfs to the whole model and the degree of confidence one has in each pdf estimate based on the number of observations of that context. This deleted estimation procedure (named so because some of the training data has been deleted) requires large amounts of computation. Carefully considered constraints must be placed on the final set of weights in order that they will exhibit the desired behavior.

The third option of using recognition experiments to determine the weights should theoretically result in the best performance. However, the computation needed is indeed excessive, and we have not used this method except in a limited way. For the manual method, we have run a few experiments with the purpose of optimizing the weights. Small improvements can be made by this method.

5. TRAINING AND RECOGNITION SYSTEMS

In this section, we describe the algorithms used in all of the recognition experiments described below.

5.1 Analysis

Input speech is lowpassed at 10 kHz and sampled at 20 kHz. Mel-frequency cepstral coefficients (MFCC) are computed as follows. Every 10 ms, a 20 ms window of speech is multipled by a Hamming window. The log power spectrum is computed via a 512 point FFT. The log power spectrum is warped according to Mel-frequency bands, resulting in a new array. An inverse FFT is then used to produce 14 real Mel-Frequency cepstral coefficients (MFCC) for each 10 ms analysis frame. Some of the training data is used with a nonuniform binary clustering algorithm [14] to produce a representative set of MFCC vectors. Each MFCC vector in the training and test sets is then classified as one of the vectors.

5.2 Variable-Frame-Rate

To save computation, strings of up to 3 identical vector codes are compressed to 1 observation. This crude variable frame rate (VFR) compression was found to reduce all computation by a factor of 2 with no loss in performance. We have not experimented with more elaborate VFR schemes.

5.3 Training

Some of the speech material to be used for training data is carefully labeled, indicating the beginning frame of each phoneme. This labeled data is used to form an initial estimate of the probability density functions (pdf) for each phoneme. Unobserved spectral probabilities are set to a low, nonzero value. Initially, the 3 pdf's in the HMM for a phoneme are all assumed to be equal. The transition probabilities emanating from each node are also assumed to be equal. Finally, all the context-dependent HMM's for each phoneme are set equal to the single, context-independent model for each phoneme.

The remainder of the training data is transcribed with a phonetic sequence (no time labels). The complete set of training data is then processed with the Forward-Backward algorithm. In each pass of the Forward-Backward algorithm the parameters of the models are updated such that the probability of training sequences given the model increases. The program is run over the entire training set until this estimated probability has started to converge. This typically requires 5 to 6 passes.

Due to the large number of models being trained, dozens of difficult implementation details had to be solved in order for the program to be able to run for a single sentence at a time on a VAX 11/780 in a reasonable amount of time (CPU time equal to 4 times the speech time), and with a virtual memory limitation of 12 Megabytes.

After each pass of the forward-backward algorithm, each of the pdf's is reestimated and the low values are again clipped to avoid the problem of probabilities equal to zero. The clipping value used depends on the number of spectra in the pdf and the amount of training data.

5.4 Recognition

Once the HMM's have been fully trained, the recognition experiments can be performed. The recognition program attempts to find the sequence of phoneme models that are most likely given the observed sequence of spectra in a test utterance. In addition to the information in the pdf's, we also have information regarding the relative likelihood of different phoneme sequences. Thus, using Bayes' rule:

$$p(\text{phoneme seq} \mid \text{spectra}) = p(\text{phoneme seq}) \times \frac{p(\text{spectra} \mid \text{phoneme seq})}{p(\text{spectra})} \qquad (1)$$

Assuming that the probability of the phoneme sequence can be modeled as a first order Markov chain, and that the pdf's of the spectra during a phoneme depend only on the phoneme and the two neighboring phonemes, this can be rewritten as:

$$p(\text{ph seq}|\text{spectra}) = \prod_{\text{all ph}} p(ph_i|ph_{i-1}) \times \frac{p(\text{sp in i}|ph_{i-1},ph_i,ph_{i+1})}{p(\text{spectra in } ph_i)} \qquad (2)$$

where "sp in i" means the spectra that occur in phoneme i.

One might think that the above statement and sufficient computing would then result in a well-defined "optimal" answer. However, one can clearly see that this estimate of the probability of a phoneme sequence decreases as more phonemes are added to the theory. This is due to the the Markov assumption and the incomplete set of acoustic features used. Thus there must be some ad hoc terms added to allow us to determine how many phonemes there are in the sequence. It is also not clear how "important" the phoneme sequence information is relative to the spectral information. There is no reason to believe that relative importance assigned by the above equation is correct. For example, if we had analyzed the speech using a 5 ms frame shift instead of a 10 ms frame shift, there would be twice as many spectral terms for the same number of phoneme sequence terms. Therefore, we use two ad hoc factors to solve this problem. Each phoneme sequence probability (first order Markov probability) is divided by an "offset" roughly equal to a higher-than-average value (somewhere in the range of 0.1 to 0.2). Then, the divided probability is raised to a power (usually in the range 1.5 to 2) to make it of proper importance relative to the spectral pdf's. The particular optimal values for these parameters are determined empirically to result in the proper balance between deleted and inserted phonemes, while maintaining the highest possible performance. Thus the equation is now:

$$p(\text{ph seq}|\text{sp}) = \prod_{\text{all ph}} \left(\frac{p(ph_i|ph_{i-1})}{\text{offset}} \right)^{\text{power}} \times \frac{p(\text{sp in i}|ph_{i-1},ph_i,ph_{i+1})}{p(\text{spectra in } ph_i)} \qquad (3)$$

5.5 Search Strategies

There are several possible search strategies that one may use to find the most likely sequence of phonemes in a sentence. The two that we have considered are a "best-first" search strategy and a time synchronous "beam search".

5.5.1 Best-First Search

The best-first strategy maintains a list or stack (usually a tree) of theories for different phoneme sequences. Then, given an evaluation criterion, it advances the most promising theory by all possible next phonemes. The updated theories (which are now longer by one phoneme) are replaced on the stack and the procedure is continued until a complete theory is found. If the evaluation criterion is non-increasing - that is, if adding another phoneme to a theory is guaranteed not to increase the theory score - as with probabilities, then the simple algorithm can be used. This algorithm always extends the best theory. When the best theory spans the whole utterance, it can be guaranteed to be the best answer (according to the scoring function). Unfortunately, this algorithm suffers from "thrashing", since it continually switches theories because extending a theory usually brings it below some other shorter theories. Theory normalization procedures must be used to properly weigh the score and the length of the different theories. One must be careful to "merge" theories that have arrived at the same time in the utterance with the same constraints as to the following theories. This general strategy has the theoretical advantage that it saves computation because it spends effort where it appears to be most likely to pay off. However, in the case of phoneme recognition, the overhead incurred is very expensive, and this may not be the best strategy.

Implicit in the notion of a theory is the concept of when the last phoneme in that theory has ended. The computation of alignments of spectral sequences through the HMM for a phoneme must be performed until there is little chance that the phoneme could extend any further. Since phonemes are relatively short events, the extra computation is large compared to the length of the phoneme. The decision of when the phoneme is over is complicated and is prone to error.

The very nature of the best-first search requires that theories of unequal length must be compared. This is difficult to do, since the different theories do not contain the same amount of evidence. While ad hoc factors can make the comparison easier, it is always necessary to consider many unlikely theories "just in case" they will turn out to be the best.

The coordinated stack and tree data structures that are necessary for the best-first search are quite complicated, compared to the simple dynamic programming involved in comparing a single model to the spectral sequence.

Finally, since the decision of which theory to process next is inherently sequential, it is difficult to get large improvements in speed by the use of large numbers of multiple processors. The simple algorithm of always processing the first several theories on the stack in parallel will become inefficient when the number of processors is more than a few.

5.5.2 Time-Synchronous Beam-Search

The time-synchronous beam search considers the set of phoneme models as being combined in one huge HMM. For each model there are five states, three of which must be "updated" for every observed spectrum. This means that each model will also propagate a score to every possible following model in each frame. Whenever two or more paths come together within a phoneme model, their probabilities are added. However, when two different phoneme models propagate to the same other phoneme model, only the best is remembered. The dynamic programming process is performed until the end of an utterance or until silence has been verified. Then, the best sequence of phoneme models through the large HMM is determined and reported. In principle, this approach considers all possible sequences of phoneme models.

The process described above is most similar to the "one-pass" dynamic programming process used for connected word matching by Bridle [15] and Jouvet [16]. However, there is one theoretical difference that must be considered. Since we add the contributions of different paths coming into the same state of the HMM, rather than keeping only the best path, we cannot guarantee that this procedure will always find the most likely sequence of models. The process we describe is somewhere in between the Viterbi process and the correct solution of the most likely sequence of models. We have found, however, that the answers that result always are the same in all three cases.

Now, let us consider the merits of this time-synchronous process. First, the computation involved is extremely simple and repetitive. It consists simply of updating the many thousands of states in all the models. Therefore, it is quite easy to program on an array processor or a microprocessor chip. In principle, it finds an "almost optimal" sequence, without having to worry about whether the search was conservative enough. If a suboptimal search is desired, it is quite easy to prune out most of the theories simply because they are sufficiently below the best theory in the frame. All comparisons of different theories are based on the same amount of input information, and thus a very tight pruning threshold is possible without changing the final answer.

One can easily verify, simply by varying the pruning threshold, what the relation is between the threshold and the probability of getting a different answer than the optimal search.

Because the process is time-synchronous, a simple reordering of all the pdf's by spectrum (rather than by model) makes it possible to deal with a small subset of the data (probabilities for the HMM's), at any time. Thus, in our case, where the pdf's are stored on a disk, the amount of disk I/O is greatly reduced. Since the recognition is performed left to right, the delay from the end of an utterance to an answer is small and fixed.

Finally, since the pruning procedure is based on a single global threshold for each frame, it is possible to have as many independent processors as desired, each working on different models, and still get savings proportional to the number of processors. Of course, care must be taken that the models in a given processor are maximally different so that all processors will be doing about the same amount of work.

6. EXPERIMENTS WITH E-SET

In this section we describe a set of preliminary experiments performed to illustrate the effect of using phonetic context. The E-Set consists of the 9 letters of English that end with the phoneme /i/: B,C,D,E,G,P,T,V,Z. A single speaker said each of the 9 letters in isolation. The set of 9 letters was repeated 40 times. The speech was analyzed as described above, and each spectrum was represented as one of 64 MFCC vectors. At this point we would like to emphasize that the particular numerical results cited here are not of any importance in themselves. Rather, the differences in performance between experimental conditions is indicative of the usefulness of the algorithms suggested.

For the E-Set, the models for the consonants do not depend on phonetic context, since they always appear preceded by silence, and followed by /i/. The /i/ phoneme, however, appears with 9 different left contexts. We call the model that depends on the phoneme to the left the "left-context" model.

Figure 2 shows the results of the three experiments. The horizontal axis represents the amount of training speech used (tokens per letter). The circles show the performance where only phoneme models are used. In this case, with one token of each letter, there are 9 tokens of the phoneme /i/. The squares show the performance where a separate model is used for each left context. For one token of each letter, there is only one token of each context-dependent model of /i/. As can be seen, with only 1 token per letter, the context model performs very poorly (61%), while the corresponding experiment using the phoneme model (with 9 tokens of /i/), achieves 79%. The poor results of the context model can be attributed to the fact that, with 1 token, it is difficult to estimate a 64-bin discrete pdf, and the model for the last part of /i/ is not very dependent on the phoneme to the left. When the number of training samples is increased to 4 or 10 tokens per letter, the performance using the left context model improves rapidly to 88%, while the phoneme model performance improves to 93%. As more training is made available, the phoneme model performance doesn't improve, (presumably because 90 tokens for each pdf in /i/ is more than sufficient training). However, the context model performance continues to improve to 97% with 20 tokens per letter.

From these results, one could devise a simple algorithm that used the context model only for those cases where there were more than 10 tokens of the appropriate context. For the remaining cases, only the phoneme model would be used. If, as

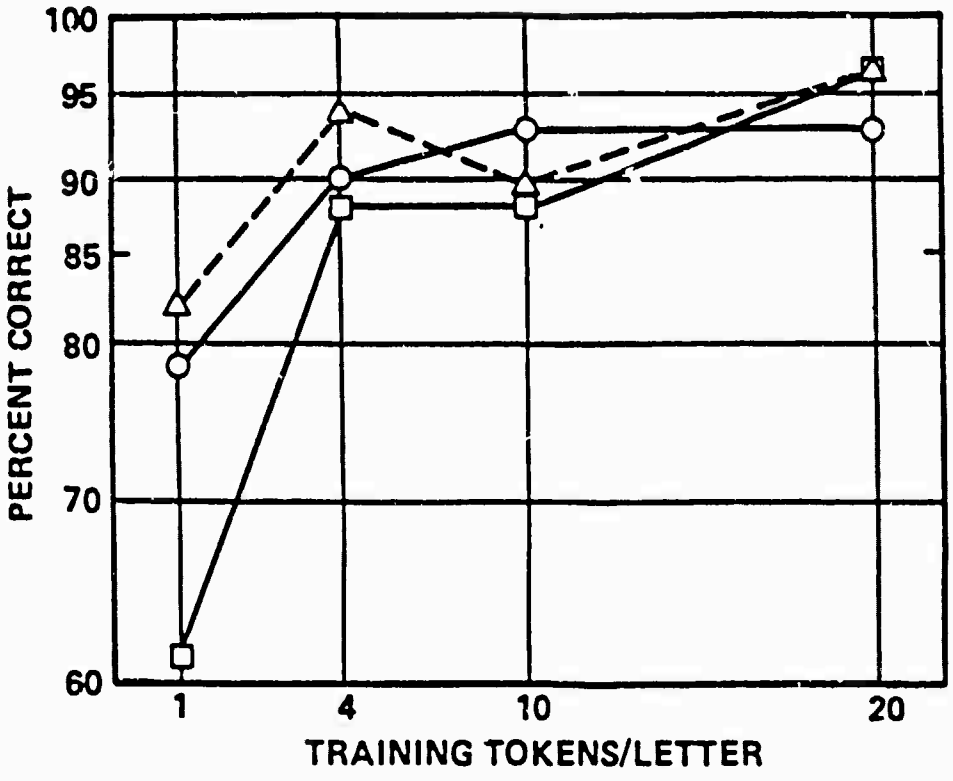

FIG. 2. E-set recognition performance

discussed in a preceding section, we allow the first part of the phoneme to depend more on the left context model, and the other parts to depend mostly on the phoneme model, the algorithm would have a lower requirement for the number of tokens for the context-dependent model for the pdf nearest the phoneme transition, and a higher requirement for the pdf farthest from the transition.

Finally, we consider the case where continuous weighting factors are used to combine the two models. The weights (which were set by hand) are dependent on the amount of training as well as the location within the phoneme, as shown in the table below:

With 1 token per letter, the context model cannot add significantly to the performance. With 4 tokens, however, the combined model outperforms either model. With 10 tokens, the performance has dropped to below that of the phoneme model alone. It appears that in this small test, the 10 tokens of training did not produce a good model for test data. With 20 tokens, there is sufficient training for the context model and, therefore, smoothing with the phoneme model does not help.

	1 Token per Letter			4 Tokens		
	Left	Mid	Right	Left	Mid	Right
Context	0.7	0.0	0.0	0.8	0.2	0.0
Phoneme	0.3	1.0	1.0	0.2	0.8	1.0

	10 Tokens			20 Tokens		
	Left	Mid	Right	Left	Mid	Right
Context	1.0	0.5	0.0	1.0	1.0	0.0
Phoneme	0.0	0.5	1.0	0.0	0.0	1.0

TABLE 1. Weight for Left Context Model

For the E-set problem, the amount of training for the different left context models is always the same. Therefore, the weights are also the same for a given experiment. However, in natural continuous speech, the frequency of different phoneme sequences varies greatly, with some sequences being several orders of magnitude more likely than others. Therefore the variation of the weights with the amount of training will become much more important.

7. EXPERIMENTS WITH CONTINUOUS SPEECH

In this section we describe several phonetic recognition experiments carried out on unconstrained continuous speech. The speech training database for continuous speech consists of 25 minutes of speech, containing a total of 550 sentences. The speech material covers three different topics: office type queries - budgets, messages, trips, etc., Harvard sentences, and children's books. The speech was digitized directly into the computer in several recording sessions spaced several days apart. 100 sentences were recorded in each session. The training material was later transcribed directly from the text without listening, using the common phonological rules for flapping, unreleased plosives, etc. 100 of the sentences were labeled carefully, with time-aligned phonemes, and used for initial statistics. 100 additional test sentences were transcribed with listening. (The purpose for transcribing the test material was so that the recognition accuracy could be determined automatically.)

Table 2 lists the phonetic recognition performance for several different configurations of the system. The table indicates which models of phonetic context were used and the corresponding phonetic recognition accuracy. The accuracy figures shown in the table were computed as the percentage of phonemes in the speech that were found in the output phoneme string in the correct place. Thus, the measure takes into account both substitutions and deletions, but not insertions. The number of insertions (which can be controlled by the phoneme sequence weight and offset) was kept constant at around 12%.

As can be seen, the models that are derived from a combination of the phoneme model and either the left or right context-dependent model resulted in significantly better performance than either the context-independent phoneme model or the left-context model alone. The system that used a combination of models dependent on left and right context simultaneously did not improve performance any further. A careful examination of the results showed that including either left context or right context produced substantially the same answers, and thus combining them did not result in any improvement. Finally, we have tried to improve the performance of the combined systems by using a deleted estimation procedure to optimize the weights for the combination. So far, this computationally expensive procedure has not resulted in any improvement over the performance when the weights are carefully selected using our intuitions about how they should vary as a function of the amount of training and the position within the phoneme.

Context Models	# spectral Templates	min. of Training	Percent Accuracy
PH	64	5	61
PH+L	64	5	71
L	64	5	51
PH+L+R	64	5	68
PH+L	128	5	75
PH	256	5	62
PH+L	256	5	75
PH+R	256	5	75
L	256	5	64
PH+L	512	5	74
PH	256	25	62
PH+L	256	25	79
L	256	25	71

TABLE 2. Phonetic recognition accuracy for continuous speech.
(PH = phoneme model; L = left-context model; R = right-context model;)

Other aspects to the results in Table 2 concern the number of spectral templates used to represent the whole spectral space for the speaker, and the amount of speech data available for training purposes. With only five minutes of training, performance improved as the number of spectral templates increased from 64 to 256. However, the performance dropped when the number increased to 512 spectral templates. This drop in performance may be an indication that the amount of training data was not sufficient for the 512-template case. Increasing the amount of training to 25 minutes, we see from Table 2 that, for 256 spectral templates, performance improved, especially for the case when the phoneme model is combined with the left-context model, to a high of 79% phonetic accuracy, with 12% insertions.

8. ACOUSTIC-PHONETIC FEATURES

As mentioned in the introduction, it is appealing to consider features other than the sequence of spectra in distinguishing phonemes. These specialized acoustic-phonetic features are computed once over a region imputed to correspond to a particular phoneme. While most such acoustic-phonetic features that may be proposed are, in fact, usually derived from the sequence of speech spectra, there is still a theoretical advantage to using them. An automatic (blind) training procedure could be expected to discover the underlying distinguishing characteristics, given enough training data. However, the amount of training data needed is prohibitively large. If the human researcher, through some special insight gained as a result of an understanding of the underlying process, can supply the recognition system with the important dimensions, a much smaller amount of training data will be sufficient to derive accurate and effective pdf's for discrimination among these phonemes. However, since a set of acoustic-phonetic features supplied by a human researcher is not usually sufficient to completely describe the speech, it is advantageous to use the spectral representation as a foundation on which to build specific acoustic-phonetic distinctions.

Therefore, we have designed a system in which we can combine both the spectral pdf information, and any acoustic-phonetic features that the researcher cares to specify, into a single HMM formalism.

After the three spectral pdf states, there is a sequence of acoustic-phonetic feature states. The number of states will be different for different phonemes, and the features that are measured in each will also vary with the particular distinctions being made. The state actually contains a pointer to a function that will compute the desired features, and a set of pdf's for the phonemes or classes that are involved. The states are compiled from a text file that indicates how each set of features are to be used. The initial pdf for the set of features is derived using the Acoustic-Phonetic Experiment Facility (APEF) [17]. APEF allows a researcher to develop acoustic-phonetic features for large databases of labeled speech in a highly interactive manner. Once the initial pdf's are specified, the forward-backward algorithm is be used to train the pdf's on the same larger database used to train the spectral pdf's.

The features that can be used fit more closely the type of distinctions with which a phonetician might be familiar. For example, the features of voice-onset-time and low-frequency energy during a plosive can be used to decide between voiced and

unvoiced plosives. Although there are only *two* 2-Dimensional pdf's in this case, the HMM is automatically compiled to contain pointers within each of the 6 plosives to the appropriate pdf.

Since the number and kind of features used for different phonemes are different, it is essential that a careful normalization of the contributions of these features be included. The number computed for each of these feature states is therefore:

$$\frac{p(\text{phoneme}\,|\,\text{features})}{p(\text{phoneme}\,|\,\text{phone class})} = \frac{p(\text{features}\,|\,\text{phoneme})}{p(\text{features}\,|\,\text{phone class})} \qquad (4)$$

That is, given a phone class (for example, unvoiced plosives), we compute an adjustment to the conditional probability that the phoneme is /p/, /t/, or /k/, using only features relevant to that distinction. After each of these adjustments (which are numbers that can be greater or less than 1) are computed for each phoneme, they are simply multiplied together with the probability coming out of the spectral HMM. While this procedure doesn't make sense in terms of a rigorous formulation of probabilities, we have found empirically that these probability adjustments do improve the separation among the phonemes intended, with little or no effect on any other phonemes.

At this time we have implemented acoustic-phonetic features for a small number of phonetic distinctions. Each one typically can decrease the errors among the chosen phonemes by as much as a factor of 2. For example, while the spectral HMM system correctly identified 61% of the unvoiced plosives as /ptk/, when the features were added, the performance among these three phonemes went up to 85%.

The results given in the table in the previous section do not include the use of acoustic-phonetic features, as we have only implemented acoustic-phonetic features for a small number of phonetic distinctions. The process of finding these features for the many phonetic distinctions that need to be made will require a large amount of human effort, and therefore is expected to take a long time to complete.

9. CONCLUSION

We have described our progress in developing techniques for phonetic recognition in unrestricted continuous speech. Our method, based on a context-dependent phonetic hidden Markov model, automatically uses information about adjacent phonemes only to the extent that it has seen examples of that context in training, and combines this information with less context-specific models for the phoneme. The combined model is shown to result in better performance than either model by itself. We have also observed that increasing the number of spectral templates from 64 to 256 and the amount of training data from 5 to 25 minutes has resulted in improved recognition. We have also devised a formalism that allows us to combine the generally useful spectral pdf information with more specifically designed acoustic-phonetic features in order to improve the discrimination power among particular phonemes. The inclusion of acoustic-phonetic features has made substantial improvements in the distinction among the phonemes for which they were intended.

In the future, our research will cover two main areas. First, we must develop robust methods to combine the models conditioned on left context with models conditioned on right context. Second, there is still much work to be done with Acoustic-Phonetic features. We need to adapt our context-dependent modeling techniques to these features, and in addition, we must find many sets of acoustic-phonetic features to improve the many possible minimal pair distinctions.

References

1. J. Makhoul, R. Schwartz, C. Cook, and D. Klatt, "A Feasibility Study of a Very Low Rate Speech Compression System", Final Report No. 3508, Bolt Beranek and Newman Inc., February 1977, Contract No. MDA903-75-C-0180, AD-A044400.

2. W.A. Woods, M. Bates, G. Brown, B. Bruce, C. Cook, J. Klovstad, J. Makhoul, B. Nash-Webber, R. Schwartz, J. Wolf, and V. Zue, "Speech Understanding Systems:", Final Report No. 3438, Bolt Beranek and Newman Inc., December 1976, Vol. 1-5, Contract N00014-75-C-0533.

3. L.R. Bahl, F. Jelinek, and R.L. Mercer, "A Maximum Likelihood Approach to Continuous Speech Recognition", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. PAMI-5, No. 2, March 1983, pp. 179-190.

4. L.R. Bahl, R. Bakis, P.S. Cohen, A.G. Cole, F. Jelinek, B.L. Lewis, and L. Mercer, "Recognition Results with Several Experimental Acoustic Processors", *IEEE Int. Conf. Acoust., Speech, Signal Processing*, Washington, DC, April 1979, pp. 249-251.

5. N.R. Dixon and H.F. Silverman, "The 1976 Modular Acoustic Processor (MAP)", *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-25, No. 5, October 1977, pp. 367-379.

6. R. Schwartz, J. Klovstad, J. Makhoul, and J. Sorensen, "A Preliminary Design of a Phonetic Vocoder Based on a Diphone Model", *IEEE Int. Conf. Acoust., Speech, Signal Processing*, Denver, CO, April 1980, pp. 32-35, Vol. 1.

7. A.E. Rosenberg, L.R. Rabiner, S.E. Levinson, and J.G. Wilpon, "A Preliminary Study on the Use of Demisyllables in Automatic Speech Recognition", *IEEE Int. Conf. Acoust., Speech, Signal Processing*, Atlanta, GA, April 1981, pp. 967-970.

8. O. Fujimura, "The Syllable as a Unit of Speech Recognition", *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-23, No. 1, February 1975, pp. 82-87.

9. M.J. Hunt, M. Lennig, and P. Mermelstein, "Experiments in Syllable-Based Recognition of Continuous Speech", *IEEE Int. Conf. Acoust., Speech, Signal Processing*, Denver, CO, April 1980, pp. 880-883, Vol. 3.

10. J.K. Baker, "The Dragon System -- An Overview", *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-23, No. 1, February 1975, pp. 24-29.

11. R. Schwartz, Y. Chow, S. Roucos, M. Krasner, and J. Makhoul, "Improved Hidden Markov Modeling of Phonemes for Continuous Speech Recognition", *IEEE Int. Conf. Acoust., Speech, Signal Processing*, San Diego, CA, March 1984, pp. 35.6.1-35.6.4, Vol. 3, Paper 35.6.

12. L.E. Baum and J.A. Eagon, "An Inequality with Applications to Statistical Estimation for Probabilistic Functions of Markov Processes and to a Model of Ecology", *Amer. Math Soc. Bulletin*, Vol. 73, 1967, pp. 360-362.

13. G.D. Forney, Sr., "The Viterbi Algorithm", *Proc. IEEE*, Vol. 61, 1973, pp. 268-278.

14. S. Roucos, R. Schwartz, and J. Makhoul, "Vector Quantization for Very-Low-Rate", *IEEE Global Telecommunications Conf.*, Miami, FL, November 29 - December 2 1982, pp. 1074-1078. Vol. 3.

15. J.S. Bridle and M.D. Brown, "Connected Word Recognition Using WholeWord Templates", *Proc. of the Inst. of Acoustics*, Autumn 1979.

16. D. Jouvet and R. Schwartz, "One-Pass Syntax-Directed Connected-Word Recognition in a Time-Sharing Environment", *IEEE Int. Conf. Acoust., Speech, Signal Processing*, San Diego, CA, March 1984, pp. 35.8.1-35.8.4, Vol. 3, Paper 35.8.

17. R.M. Schwartz, "Acoustic-Phonetic Experiment Facility for the Study of Continuous Speech", *IEEE Int. Conf. Acoust., Speech, Signal Processing*, Philadelphia, PA, April 1976, pp. 1-4.